

Руководство администратора Uncom OS

Важно:

Роль администратора Uncom OS подразумевает навыки работы с операционными системами на базе Linux, семейства Debian.

Отличительной особенностью Uncom OS является её дружелюбность к пользователю и минимальные требования его к специальным навыкам. Администрирование Uncom OS не отличается от администрирования Linux Debian.

Инструкция администратора

1. Прочитать «Руководство пользователя операционной системы Uncom OS»
2. Пройти обучение по администрированию операционных систем Linux, например, в компании разработчике Uncom OS, заявку можно оставить по адресу info@uncom.tech
3. Если первые пункта инструкции не помогли решить проблему, следует обратиться в службу поддержки по адресу helpdesk@uncom.tech (для подтверждения права на получение поддержки, может быть запрошен ключ активации)

Приложение

Для администраторов, прошедших обучение, или желающих пройти его самостоятельно публикуем руководство администратора операционной системы Uncom OS, основанной на Linux.

Добро пожаловать в руководство администратора операционной системы Uncom OS, основанной на Linux. В этом руководстве, мы предоставим вам расширенное руководство по управлению системой на уровне администратора.

Глава 1: Введение в Linux

Linux - это операционная система с открытым исходным кодом, которая активно используется в мире серверов и настольных компьютеров. В этой главе вы узнаете о базовых концепциях Linux, включая структуру файловой системы, права доступа и командную строку.

Глава 2: Установка и настройка

В этой главе мы рассмотрим процесс установки Linux и настройки системы для вашего сервера или ПК. Мы также покажем вам, как настроить сеть, создать пользователей и установить программное обеспечение.

Глава 3: Управление пакетами

В Linux пакеты - это способ управления программным обеспечением. В этой главе мы рассмотрим, как устанавливать, обновлять и удалять пакеты в Linux.

Глава 4: Управление файловой системой

Файловая система Linux содержит все файлы и директории, которые хранятся на вашем сервере или ПК. В этой главе мы рассмотрим, как управлять файловой системой, включая создание, перемещение и удаление файлов и директорий.

Глава 5: Управление сервисами

Linux запускает множество сервисов, которые обеспечивают работу вашего сервера или ПК. В этой главе мы рассмотрим, как управлять сервисами, включая запуск, остановку и перезапуск.

Глава 6: Безопасность

Безопасность - это очень важный аспект управления сервером или ПК. В этой главе мы рассмотрим, как обеспечить безопасность вашего сервера или ПК, включая настройку брандмауэра, управление доступом и настройку безопасности приложений.

Глава 7: Управление резервными копиями

Резервное копирование - это важная часть управления сервером или ПК. В этой главе мы рассмотрим, как создавать и восстанавливать резервные копии данных на вашем сервере или ПК.

Глава 8: Управление производительностью

В этой главе мы рассмотрим, как управлять производительностью вашего сервера или ПК, включая настройку ресурсов и оптимизацию производительности.

Глава 9: Управление ошибками и журналированием

В этой главе мы рассмотрим, как управлять ошибками и журналированием на вашем сервере или ПК. Мы также рассмотрим, как использовать инструменты для анализа журналов и устранения ошибок.

Глава 10: Заключение

В этой главе мы подведем итоги и предоставим вам рекомендации по администрированию операционной системы Uncom OS.

Глава 1: Введение в Linux

Linux - это операционная система с открытым исходным кодом, которая активно используется в мире серверов и настольных компьютеров. Она была создана в 1991 году Линусом Торвальдсом и стала одной из самых популярных операционных систем в мире.

Структура файловой системы

Linux использует иерархическую файловую систему, что означает, что все файлы и директории находятся внутри корневой директории. Корневая директория обозначается символом "/", и все остальные директории строятся относительно нее. Она имеет следующую структуру:

- /bin: содержит исполняемые файлы, которые используются системой и пользователями
- /boot: содержит файлы, необходимые для загрузки системы
- /dev: содержит файлы, представляющие устройства, подключенные к системе
- /etc: содержит конфигурационные файлы системы и приложений
- /home: содержит домашние каталоги пользователей
- /lib: содержит библиотеки, используемые системой и приложениями
- /media: содержит точки монтирования для съемных носителей
- /mnt: содержит точки монтирования для временных файловых систем
- /opt: содержит дополнительные приложения
- /proc: содержит информацию о процессах в системе
- /root: домашняя директория суперпользователя
- /run: содержит файлы временных данных для текущего сеанса
- /sbin: содержит исполняемые файлы, используемые системным администратором
- /srv: содержит данные, используемые серверами
- /sys: содержит информацию о системе и устройствах
- /tmp: содержит временные файлы
- /usr: содержит основные приложения и библиотеки
- /var: содержит переменные данные, включая журналы и кэши

Права доступа

Linux использует систему прав доступа, которая определяет, какие пользователи и группы могут получить доступ к файлам и директориям. Каждый файл и директория имеют три типа прав доступа: чтение, запись и исполнение. Права доступа могут быть установлены для владельца файла, группы и других пользователей. Владелец файла имеет наивысшие права доступа, а другие пользователи могут иметь только часть прав.

Командная строка

Командная строка Linux - это мощный инструмент для управления системой. Она позволяет запускать команды и скрипты, управлять файловой системой, устанавливать программное обеспечение и многое другое. В этой главе вы узнаете основы работы с командной строкой Linux, включая:

- Основные команды, такие как ls, cd и pwd
- Как работать с файлами и директориями, включая создание, перемещение и удаление

- Как управлять процессами и службами
- Как устанавливать и удалять программное обеспечение с помощью пакетных менеджеров
- Как работать с текстовыми файлами и настраивать текстовые редакторы

Более подробную информацию о каждой из этих тем вы можете найти в соответствующих главах нашего руководства.

Глава 2: Установка и настройка

Установка и настройка Linux - это первый шаг в управлении вашим сервером или ПК. В этой главе мы рассмотрим, как установить Linux на ваш сервер или ПК и настроить его для работы.

Установка Linux

После того, как вы выбрали дистрибутив, вы можете приступить к установке Linux на ваш сервер или ПК. Установка может быть выполнена различными способами, в зависимости от дистрибутива, который вы выбрали.

Настройка Linux

После установки Linux вы можете приступить к его настройке. Настройка Linux может включать в себя:

- Создание пользователей и групп
- Настройка сетевых подключений
- Настройка брандмауэра
- Настройка SSH
- Настройка времени и даты
- Настройка управления пакетами

Каждый дистрибутив Linux имеет свои собственные инструменты для настройки системы, и вы должны ознакомиться с документацией для вашего дистрибутива, чтобы узнать, как выполнить настройку.

Обновление Linux

После установки и настройки Linux вы должны регулярно обновлять систему, чтобы получать последние исправления и функции. Обновление Linux может быть выполнено с помощью пакетных менеджеров, таких как apt или yum.

Резервное копирование

Резервное копирование - это важная часть управления сервером или ПК. Вы должны регулярно создавать резервные копии данных на вашем сервере или ПК, чтобы защитить их от потери или повреждения. Резервное копирование может быть выполнено с помощью различных инструментов, таких как rsync или tar.

Более подробную информацию о каждой из этих тем вы можете найти в соответствующих главах нашего руководства.

Глава 3: Управление пакетами

Управление пакетами - это процесс установки, обновления и удаления программного обеспечения на сервере или ПК. В Linux для этого обычно используются пакетные менеджеры.

Пакетные менеджеры

Пакетные менеджеры - это инструменты, которые позволяют устанавливать, обновлять и удалять программное обеспечение на сервере или ПК. Они также позволяют управлять зависимостями пакетов и контролировать версии программного обеспечения. В Uncom OS пакетные менеджеры включают в себя:

- `apt-get` (Debian, Ubuntu, Uncom OS)

Установка пакетов

Установка пакетов - это процесс установки новых программ на сервер или ПК. Для установки пакета можно использовать команду `install` вашего пакетного менеджера. Например, для установки пакета `nginx` в Uncom OS, можно использовать следующую команду:

```
sudo apt-get install nginx
```

Обновление пакетов

Обновление пакетов - это процесс обновления установленного программного обеспечения до последней версии. Для обновления пакетов можно использовать команду `update` вашего пакетного менеджера, чтобы загрузить новые списки пакетов, а затем команду `upgrade`, чтобы обновить установленные пакеты. Например, для обновления всех установленных пакетов в Uncom OS, можно использовать следующие команды:

```
sudo apt-get update  
sudo apt-get upgrade
```

Удаление пакетов

Удаление пакетов - это процесс удаления установленного программного обеспечения с сервера или ПК. Для удаления пакета можно использовать команду `remove` вашего пакетного менеджера. Например, для удаления пакета `nginx` в Uncom OS, можно использовать следующую команду:

```
sudo apt-get remove nginx
```

Поиск пакетов

Поиск пакетов - это процесс поиска пакета в репозитории вашего пакетного менеджера. Для поиска пакета можно использовать команду `search` вашего пакетного менеджера.

Например, для поиска пакета `nginx` в Uncom OS, можно использовать следующую команду:

```
sudo apt-cache search nginx
```

Управление зависимостями

Зависимости пакетов - это другие пакеты, которые требуются для работы определенного программного обеспечения. Пакетные менеджеры позволяют автоматически устанавливать зависимости пакетов при установке новых программ. Также можно использовать команду `depend` вашего пакетного менеджера, чтобы узнать, какие пакеты требуются для работы определенного программного обеспечения.

Для более подробной информации о каждой из этих тем, вы можете обратиться к соответствующим главам нашего руководства.

Глава 4: Управление файловой системой

Управление файловой системой - это процесс управления файлами и папками на сервере или ПК. В Linux файловая система организована в виде древовидной структуры, в которой каждый узел представляет собой файл или папку. Настройка файловой системы является важной частью администрирования Linux. В этой главе мы рассмотрим основные команды для управления файловой системой.

Просмотр файлов и папок

Для просмотра содержимого текущей директории можно использовать команду `ls`. Например, для просмотра списка файлов и папок в текущей директории, можно использовать следующую команду:

```
ls
```

Эта команда выведет список файлов и папок из текущей директории. Для просмотра списка файлов и папок в другой директории, нужно указать путь к ней. Например, для просмотра списка файлов и папок в директории `/var/log`, можно использовать следующую команду:

```
ls /var/log
```

Создание файлов и папок

Для создания нового файла можно использовать команду `touch`. Например, для создания нового файла `example.txt`, можно использовать следующую команду:

```
touch example.txt
```

Эта команда создаст новый файл `example.txt` в текущей директории. Для создания новой папки можно использовать команду `mkdir`. Например, для создания новой папки `example_folder`, можно использовать следующую команду:

```
mkdir example_folder
```

Эта команда создаст новую папку `example_folder` в текущей директории.

Перемещение и переименование файлов и папок

Для перемещения файла или папки на другое место можно использовать команду `mv`. Например, для перемещения файла `example.txt` в папку `/var/log`, можно использовать следующую команду:

```
mv example.txt /var/log
```

Эта команда переместит файл `example.txt` из текущей директории в папку `/var/log`. Для переименования файла или папки можно использовать команду `mv`. Например, для

переименования файла `example.txt` в `new_example.txt`, можно использовать следующую команду:

```
mv example.txt new_example.txt
```

Эта команда переименует файл `example.txt` в `new_example.txt`.

Копирование файлов и папок

Для копирования файла или папки можно использовать команду `cp`. Например, для копирования файла `example.txt` в папку `/var/log`, можно использовать следующую команду:

```
cp example.txt /var/log
```

Эта команда скопирует файл `example.txt` из текущей директории в папку `/var/log`. Для копирования папки со всем содержимым можно использовать опцию `-r`. Например, для копирования папки `example_folder` в `/var/log`, можно использовать следующую команду:

```
cp -r example_folder /var/log
```

Эта команда скопирует папку `example_folder` и все ее содержимое в папку `/var/log`.

Удаление файлов и папок

Для удаления файла можно использовать команду `rm`. Например, для удаления файла `example.txt`, можно использовать следующую команду:

```
rm example.txt
```

Эта команда удалит файл `example.txt`. Для удаления папки со всем содержимым можно использовать опцию `-r`. Например, для удаления папки `example_folder`, можно использовать следующую команду:

```
rm -r example_folder
```

Эта команда удалит папку `example_folder` и все ее содержимое.

Изменение прав доступа к файлам и папкам

Для изменения прав доступа к файлу или папке можно использовать команду `chmod`. Например, для установки прав доступа `read`, `write` и `execute` на файл `example.txt` можно использовать следующую команду:

```
chmod 700 example.txt
```

Эта команда установит права доступа `read`, `write` и `execute` для владельца файла `example.txt`, и не даст никаких прав доступа для группы и остальных пользователей.

В этой главе мы рассмотрели основные команды для управления файловой системой в Linux. Эти команды помогут вам управлять файлами и папками на сервере или ПК и настроить файловую систему для вашего приложения или сайта.

Глава 5: Управление сервисами

Управление сервисами - это процесс управления службами, которые работают на сервере или ПК. Службы могут быть как системными, так и разработанными сторонними разработчиками. Некоторые из служб могут работать в фоновом режиме и предоставлять доступ к приложениям или сайтам, а другие могут запускаться только при запросе пользователя.

Запуск, остановка и перезапуск сервисов

Для управления сервисами можно использовать команду `systemctl`. Эта команда позволяет запускать, останавливать и перезапускать службы на сервере или ПК. Например, для запуска сервиса `nginx`, можно использовать следующую команду:

```
sudo systemctl start nginx
```

Эта команда запустит службу `nginx` на сервере или ПК. Для остановки службы можно использовать команду `stop`. Например, для остановки службы `nginx`, можно использовать следующую команду:

```
sudo systemctl stop nginx
```

Эта команда остановит службу `nginx` на сервере или ПК. Для перезапуска службы можно использовать команду `restart`. Например, для перезапуска службы `nginx`, можно использовать следующую команду:

```
sudo systemctl restart nginx
```

Эта команда перезапустит службу `nginx` на сервере или ПК.

Включение и отключение автозапуска сервисов

Для включения автозапуска службы можно использовать команду `enable`. Например, для включения автозапуска службы `nginx`, можно использовать следующую команду:

```
sudo systemctl enable nginx
```

Эта команда включит автозапуск службы `nginx` при каждом запуске сервера или ПК. Для отключения автозапуска службы можно использовать команду `disable`. Например, для отключения автозапуска службы `nginx`, можно использовать следующую команду:

```
sudo systemctl disable nginx
```

Эта команда отключит автозапуск службы `nginx` при каждом запуске сервера или ПК.

Просмотр статуса сервисов

Для просмотра статуса службы можно использовать команду `status`. Например, для просмотра статуса службы `nginx`, можно использовать следующую команду:

```
sudo systemctl status nginx
```

Эта команда покажет текущий статус службы `nginx` на сервере или ПК.

Управление журналами сервисов

Для просмотра журналов службы можно использовать команду `journalctl`. Например, для просмотра журналов службы `nginx`, можно использовать следующую команду:

```
sudo journalctl -u nginx
```

Эта команда покажет журналы службы `nginx` на сервере или ПК.

В этой главе мы познакомились с основными командами для управления сервисами на сервере или ПК. Не забывайте, что каждый сервис может иметь свои собственные команды управления, которые могут отличаться от стандартных команд. Как правило, эти команды описываются в документации службы.

Глава 6: Безопасность

Безопасность - это один из самых важных аспектов управления сервером или ПК. В этой главе мы рассмотрим основные меры безопасности, которые можно применять на сервере или ПК.

Обновление системы и программного обеспечения

Один из самых важных аспектов безопасности - это правильное обновление системы и программного обеспечения. Регулярное обновление помогает закрывать уязвимости и баги, которые могут быть использованы злоумышленниками. Для обновления системы и программного обеспечения можно использовать команду `apt-get`. Например, для обновления всех установленных пакетов можно использовать следующую команду:

```
sudo apt-get update && sudo apt-get upgrade
```

Эта команда обновит все установленные пакеты на сервере или ПК.

Управление пользователями и правами доступа

Для обеспечения безопасности сервера или ПК важно правильно управлять пользователями и правами доступа. Необходимо создавать отдельных пользователей для каждого члена команды и давать им только необходимые права доступа на сервере или ПК. Например, можно создать отдельного пользователя для управления базами данных и дать ему только необходимые права доступа к базам данных.

Для управления пользователями и правами доступа можно использовать команды `adduser`, `deluser`, `usermod`, `chown` и `chmod`. Например, для добавления нового пользователя можно использовать следующую команду:

```
sudo adduser username
```

Эта команда создаст нового пользователя с именем `username`. Для удаления пользователя можно использовать команду `deluser`. Например, для удаления пользователя `username`, можно использовать следующую команду:

```
sudo deluser username
```

Эта команда удалит пользователя `username` с сервера или ПК.

Настройка брандмауэра

Брандмауэр - это программа, которая контролирует входящий и исходящий трафик на сервере или ПК. Настройка брандмауэра - это одна из важных мер безопасности, которую необходимо применять на сервере или ПК. Для настройки брандмауэра можно использовать программы `ufw` или `iptables`. Например, для настройки брандмауэра `ufw`, можно использовать следующие команды:

```
sudo ufw allow ssh
```

```
sudo ufw allow http
sudo ufw allow https
sudo ufw enable
```

Эти команды откроют порты для SSH, HTTP и HTTPS и включат брандмауэр `ufw` на сервере или ПК.

Использование SSH-ключей

SSH-ключи - это один из самых безопасных способов аутентификации на сервере. Вместо того, чтобы использовать пароль, можно использовать SSH-ключ для подключения к серверу. Для использования SSH-ключей необходимо сгенерировать ключи на клиентской машине и добавить их на сервер.

Глава 7: Управление резервными копиями

Резервное копирование данных - это один из важных аспектов управления сервером или ПК. В этой главе мы рассмотрим основные стратегии резервного копирования и как настроить автоматическое резервное копирование на сервере или ПК.

Стратегии резервного копирования

Существуют различные стратегии резервного копирования данных. Наиболее распространенными являются полное, инкрементальное и дифференциальное резервное копирование.

Полное резервное копирование

При полном резервном копировании все данные на сервере или ПК копируются в один файл. Это занимает много места на жестком диске и требует много времени для выполнения. Однако, полное резервное копирование позволяет восстановить данные на сервере или ПК целиком.

Инкрементальное резервное копирование

При инкрементальном резервном копировании только те данные, которые были изменены с момента последнего резервного копирования, копируются в отдельный файл. Это занимает меньше места на жестком диске и требует меньше времени для выполнения, чем полное резервное копирование. Однако, чтобы восстановить данные на сервере или ПК, необходимо иметь все инкрементальные резервные копии с момента последнего полного резервного копирования.

Дифференциальное резервное копирование

При дифференциальном резервном копировании только те данные, которые были изменены с момента последнего полного резервного копирования, копируются в отдельный файл. Это занимает меньше места на жестком диске и требует меньше времени для выполнения, чем полное резервное копирование. Однако, чтобы восстановить данные на сервере или ПК, необходимо иметь последний полный резервный файл и последний дифференциальный резервный файл.

Настройка автоматического резервного копирования

Для настройки автоматического резервного копирования на сервере или ПК можно использовать программы `rsync`, `tar` и `cron`. Например, для настройки ежедневного инкрементального резервного копирования можно использовать следующую команду:

```
rsync -av --delete /var/www/ /backup/daily
```

Эта команда скопирует все данные из каталога `/var/www/` в каталог `/backup/daily` с помощью `rsync`. Команда `--delete` удалит все файлы из каталога `/backup/daily`, которых уже нет в каталоге `/var/www/`.

Глава 8: Управление производительностью

Управление производительностью сервера или ПК - это один из важных аспектов управления сервером или ПК. В этой главе мы рассмотрим основные методы управления производительностью, которые можно применять на сервере или ПК.

Мониторинг производительности

Один из самых важных аспектов управления производительностью - это мониторинг производительности сервера или ПК. Необходимо контролировать загрузку ЦП, ОЗУ и дискового пространства на сервере или ПК. Для мониторинга производительности можно использовать программы `htop`, `top`, `nmon` и `iostat`. Например, для просмотра текущей загрузки ЦП и ОЗУ можно использовать программу `htop`.

Оптимизация конфигурации сервера или ПК

Оптимизация конфигурации сервера или ПК - это еще один способ управления производительностью сервера или ПК. Необходимо настроить сервер или ПК таким образом, чтобы он использовал ресурсы максимально эффективно. Для оптимизации конфигурации сервера или ПК можно использовать программу `tuning-primer.sh`. Например, для проверки конфигурации MySQL можно использовать следующую команду:

```
wget <https://raw.githubusercontent.com/BMDan/tuning-primer.sh/master/tuning-primer.sh>
chmod +x tuning-primer.sh
./tuning-primer.sh
```

Оптимизация баз данных

Оптимизация баз данных - это еще один способ управления производительностью сервера или ПК. Необходимо настроить базы данных таким образом, чтобы они использовали ресурсы максимально эффективно. Для оптимизации баз данных можно использовать программы `mysqltuner.pl` и `pt-query-digest`. Например, для проверки конфигурации MySQL можно использовать следующую команду:

```
wget <https://raw.githubusercontent.com/major/MySQLTuner-perl/master/mysqltuner.pl>
chmod +x mysqltuner.pl
./mysqltuner.pl
```

Использование кэширования

Использование кэширования - это еще один способ управления производительностью сервера или ПК. Необходимо кэшировать данные, чтобы уменьшить количество запросов к базе данных. Для кэширования можно использовать программы `memcached` и `redis`. Например, для установки и настройки `memcached` можно использовать следующие команды:

```
sudo apt-get install memcached
sudo nano /etc/memcached.conf
```


Использование CDN

Использование CDN - это еще один способ управления производительностью сервера. Необходимо использовать CDN для ускорения загрузки статических файлов. Для использования CDN можно использовать программы CloudFlare и MaxCDN. Например, для настройки CloudFlare можно использовать следующие команды:

```
sudo apt-get install apache2  
sudo nano /etc/apache2/apache2.conf
```

Глава 9: Управление ошибками и журналированием

Управление ошибками и журналирование - это важные аспекты управления сервером или ПК. В этой главе мы рассмотрим основные методы управления ошибками и журналирования, которые можно применять на сервере или ПК.

Контроль ошибок

Один из важных аспектов управления ошибками - это контроль ошибок на сервере или ПК. Необходимо контролировать ошибки на сервере или ПК, чтобы быстро обнаруживать и устранять их. Для контроля ошибок можно использовать программы `logrotate`, `syslog` и `fail2ban`. Например, для настройки `logrotate` можно использовать следующие команды:

```
sudo apt-get install logrotate
sudo nano /etc/logrotate.conf
```

Журналирование

Журналирование - это еще один способ управления ошибками. Необходимо записывать все события на сервере или ПК в журналы, чтобы легко было определить, что происходит на сервере или ПК. Для журналирования можно использовать программы `syslog`, `rsyslog` и `journalctl`. Например, для просмотра журналов событий можно использовать следующую команду:

```
journalctl -u nginx
```

Анализ журналов

Анализ журналов - это еще один способ управления ошибками. Необходимо анализировать журналы, чтобы быстро обнаруживать и устранять ошибки. Для анализа журналов можно использовать программы `grep`, `awk` и `sed`. Например, для поиска ошибок в журнале можно использовать следующую команду:

```
grep "error" /var/log/nginx/error.log
```

Управление ротацией журналов

Управление ротацией журналов - это еще один важный аспект управления сервером или ПК. Необходимо регулярно очищать журналы, чтобы они не занимали много места на диске. Для управления ротацией журналов можно использовать программы `logrotate` и `rotatelogs`. Например, для настройки ротации журналов можно использовать следующие команды:

```
sudo nano /etc/logrotate.conf
sudo nano /etc/logrotate.d/nginx
```

Настройка уведомлений об ошибках

Настройка уведомлений об ошибках - это еще один важный аспект управления сервером или ПК. Необходимо настроить уведомления, чтобы быстро получать информацию об ошибках на сервере или ПК. Для настройки уведомлений можно использовать программы `syslog`, `rsyslog` и `logwatch`. Например, для настройки уведомлений об ошибках можно использовать следующие команды:

```
sudo nano /etc/rsyslog.conf
sudo nano /etc/logwatch/conf/logwatch.conf
```

Глава 10: Заключение

В этом руководстве мы рассмотрели основные аспекты управления операционной системой, включая установку программного обеспечения, настройку безопасности, управление производительностью, управление ошибками и журналирование. Несмотря на то, что мы рассмотрели множество различных аспектов управления операционной системой, этот список не является исчерпывающим.

Администрирование операционной системы - это непрерывный процесс, который требует постоянного мониторинга и обновления. Необходимо регулярно обновлять программное обеспечение, настраивать безопасность и управлять производительностью сервера или ПК. Только так можно обеспечить надежность и безопасность операционной системы.

Спасибо за чтение этого руководства. Мы надеемся, что оно было полезным для Вас и помогло вам лучше понять, администрировать операционную систему Uncom OS.